CLAIMS

1. A method comprising:

identifying one or more characteristics of one or more elements of a media processing system; and

dynamically negotiating which element of a media processing system will perform certain media processing tasks based, at least in part, on the identified one or more characteristics of the system element(s).

2. A method according to claim 1, wherein the method is implemented by an application program interface (API) of an operating system executing on a general purpose computing system hosting the media processing system.

3. A method according to claim 2, wherein identifying the one or more characteristics of the system elements comprises:

generating a negotiation data structure in accordance with a proposed set of media processing capabilities; ;

issuing the negotiation data structure to one or more elements of a media processing system;

determining whether each of the one or more media processing system elements accepted the negotiation data structure; and

executing media processing tasks within the proposed media processing capabilities if the negotiation data structure is accepted by the elements of the media processing system.

4. A method according to claim 3, further comprising:

generating another negotiation data structure in accordance with another proposed set of media processing capabilities , if the media processing system elements did not previously accept the negotiation data structure; and

iteratively performing the issuing, determining and generating steps until the media processing system elements accept a negotiation data structure.

5. A method according to claim 3, further comprising:

generating operational data structure(s) to pass between the one or more media processing system elements to facilitate shared decoding of multimedia content between media processing system elements.

6. A method according to claim 5, wherein the data structures include at least, one or more residual difference data structures and one or more macroblock control command data structures which have fixed sizes, determined on a per-frame basis.

7. A method according to claim 5, wherein the operational data structures comprise a raw bitstream data structure, dynamically generated to pass media content from a decoder application to a hardware accelerator to facilitate decoding by the accelerator.

8. A method according to claim 3, wherein the data structure(s) include one or more auto-negotiation data structure(s).

9. A method according to claim 8, wherein the auto-negotiation data structure(s) include a connection mode data structure denoting a proposed set of required processing system capabilities that indicate capabilities needed for decoding data in a format specified by a particular media processing standard.

10. A method according to claim 8, wherein the auto-negotiation data structure(s) include a connection configuration data structure denoting a proposed split in media processing between media processing system elements.

11. A method according to claim 1, wherein identifying one or more characteristics of media processing system elements comprises:

generating one or more auto-negotiation data structure(s) denoting a proposed set of media processing system capabilities and/or a proposed split in media processing among media system elements; and

iteratively issuing the auto-negotiation data structure(s) to media processing system elements until a proposed set of media processing system capabilities and/or split in media processing is agreed upon by each element of the media processing system.

12. A storage medium comprising a plurality of executable instructions which, when executed, implement a method according to claim 1.

13. A computing system comprising:

a storage medium having stored therein a plurality of executable instructions; and

an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 1.

14. A computing system comprising:

a media processing application;

a media processing accelerator; and

an operating system, executing on the computing system, including an application program interface (API) to facilitate communication between the media processing application and the media processing accelerator, wherein the API includes auto-negotiation data structure(s) and operational data structure(s) to dynamically negotiate at least a set of media processing system capabilities and/or a split in media processing among system elements suitable to each of the media processing application and the media processing accelerator and to processing of received media content, respectively.

15. A computing system according to claim 14, the auto-negotiation data structures comprising:

a processing mode data structure, generated by the API to propose a media set of media processing capabilities and/or a split in media processing among system elements to each of the media processing application and the media processing accelerator.

16. A computing system according to claim 15, wherein the processing mode data structure is a ConnectMode data structure.

17. A computing system according to claim 14, the auto-negotiation data structures comprising:

a connection mode data structure, specifying a set of media processing system capabilities, and/or

a processing configuration data structure, generated by the API to propose a split in media processing between the media processing application and the media processing accelerator.

18. A computing system according to claim 17, wherein the processing configuration data structure comprises a ConnectConfig data structure.

19. A computing system according to claim 14, the operational data structure(s) comprising:

one or more residual difference data structures, generated by the API to pass residual difference information between the media processing application and the media processing accelerator for media processing; and

one or more control command data structures, generated by the API to pass control commands between the media processing application and the media processing accelerator.

20. A computing system according to claim 19, wherein the residual difference data structures and the control command data structures are dynamically generated when the auto-negotiation data structures have negotiated a split in media processing between the media processing application and the media processing accelerator.

21. A computing system according to claim 14, the operational data structure(s) comprising:

a raw bitstream data structure, generated by the API to pass media content in raw bitstream form from the media processing application to the media processing accelerator.

22. A computing system according to claim 21, wherein the raw bitstream data structure is dynamically generated when the auto-negotiation data structures have negotiated that the media processing accelerator will perform the media processing.

23. A computing system according to claim 14, further comprising:

a storage medium including a plurality of executable instructions; and

an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement the operating system and associated API.

24. A computing system according to claim 23, wherein the execution unit executes at least a subset of the plurality of executable instructions to implement the media processing application.

25. A storage medium comprising a plurality of executable instructions including instructions which, when executed, implement an application program interface (API) to facilitate media processing between media processing system elements, the API including one or more auto-negotiation data structure(s) dynamically generated and issued to the media processing system elements to negotiate a set of media processing capabilities and/or a split in media processing tasks between the elements of the media processing system suitable to each of the media processing system elements.

26. A storage medium according to claim 25, wherein the auto-negotiation data structures include a processing mode data structure, iteratively issued to the media processing system elements denoting an iteratively changing the proposed set of media processing capabilities until accepted by all media processing system elements.

27. A storage medium according to claim 25, wherein the auto-negotiation data structures include a processing configuration data structure, iteratively issued to the media processing system elements denoting an iteratively changing proposed split in media processing between media processing system elements.

28. A storage medium according to claim 25, wherein the API further comprises:

one or more operational data structures, to facilitate media processing between media processing system elements based, at least in part, on a negotiated set of media processing capabilities and/or split in media processing responsibility between the media processing system elements.